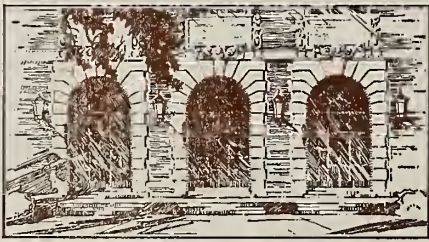


LIBRARY OF THE
UNIVERSITY OF ILLINOIS
AT URBANA-CHAMPAIGN

510.84
I l 6 r
no. 715-721
cop. 2





Digitized by the Internet Archive
in 2013

<http://archive.org/details/furthernegative721triv>

510.84
IL6N
no. 721
cop 2
UIUCDCS-R-75-721

math

Further Negative Results Regarding the Use
of Continued Fractions for Digital
Computer Arithmetic

by

Kishor Shridharbhai Trivedi

May 1975

THE LIBRARY OF THE

JUN 24 1975

UNIVERSITY OF ILLINOIS



DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN · URBANA, ILLINOIS

UIUCDCS-R-75-721

Further Negative Results Regarding the Use
of Continued Fractions for Digital
Computer Arithmetic

by

Kishor Shridharbhai Trivedi

May 1975

Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, Illinois

This work was supported in part by the National Science Foundation under Grant No. NSF DCR 73-07998.

Acknowledgment

The author wishes to thank Professor James E. Robertson for his continued support and encouragement. Thanks are also due to Mrs. June Wingler for typing this paper.

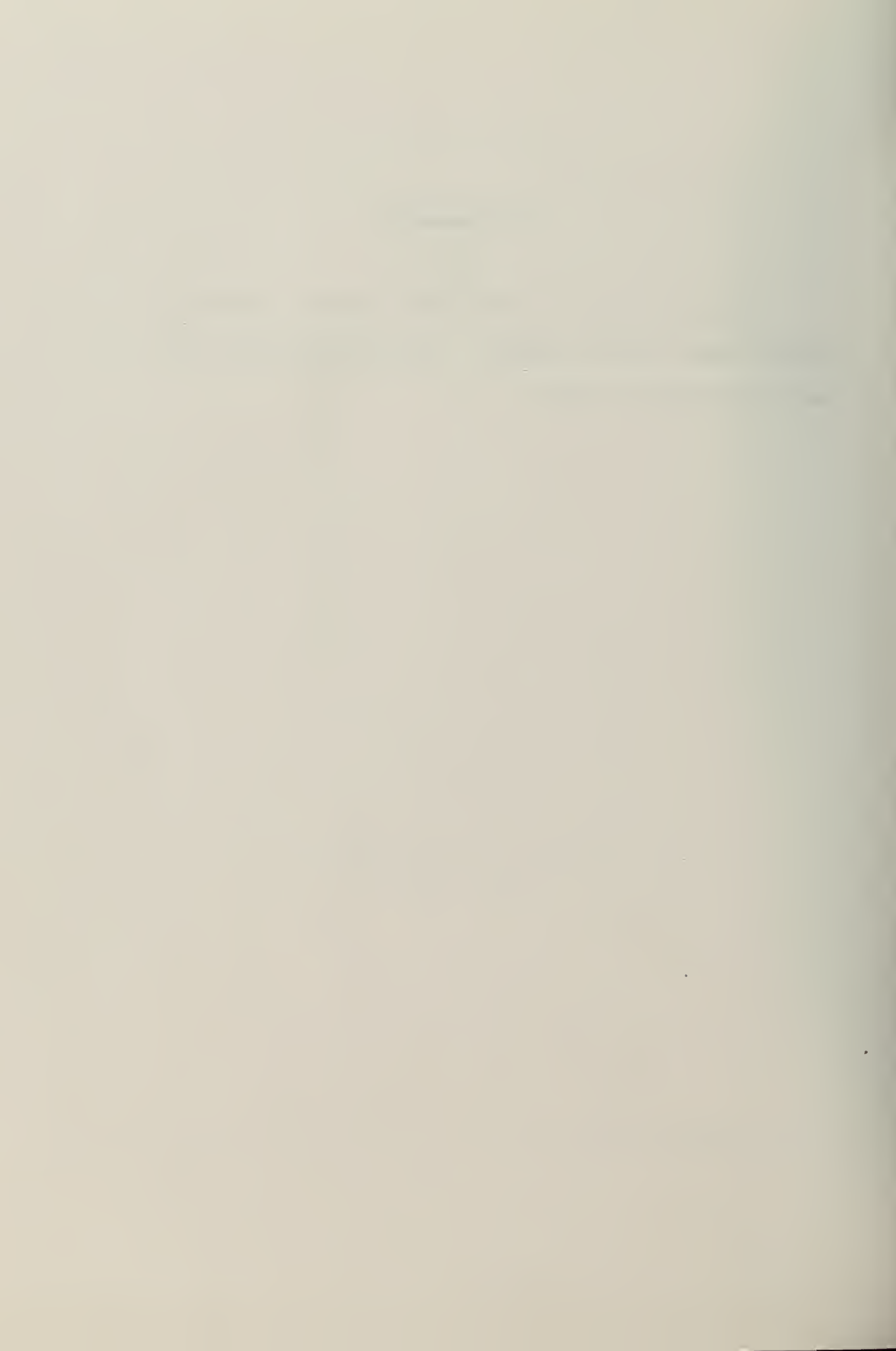
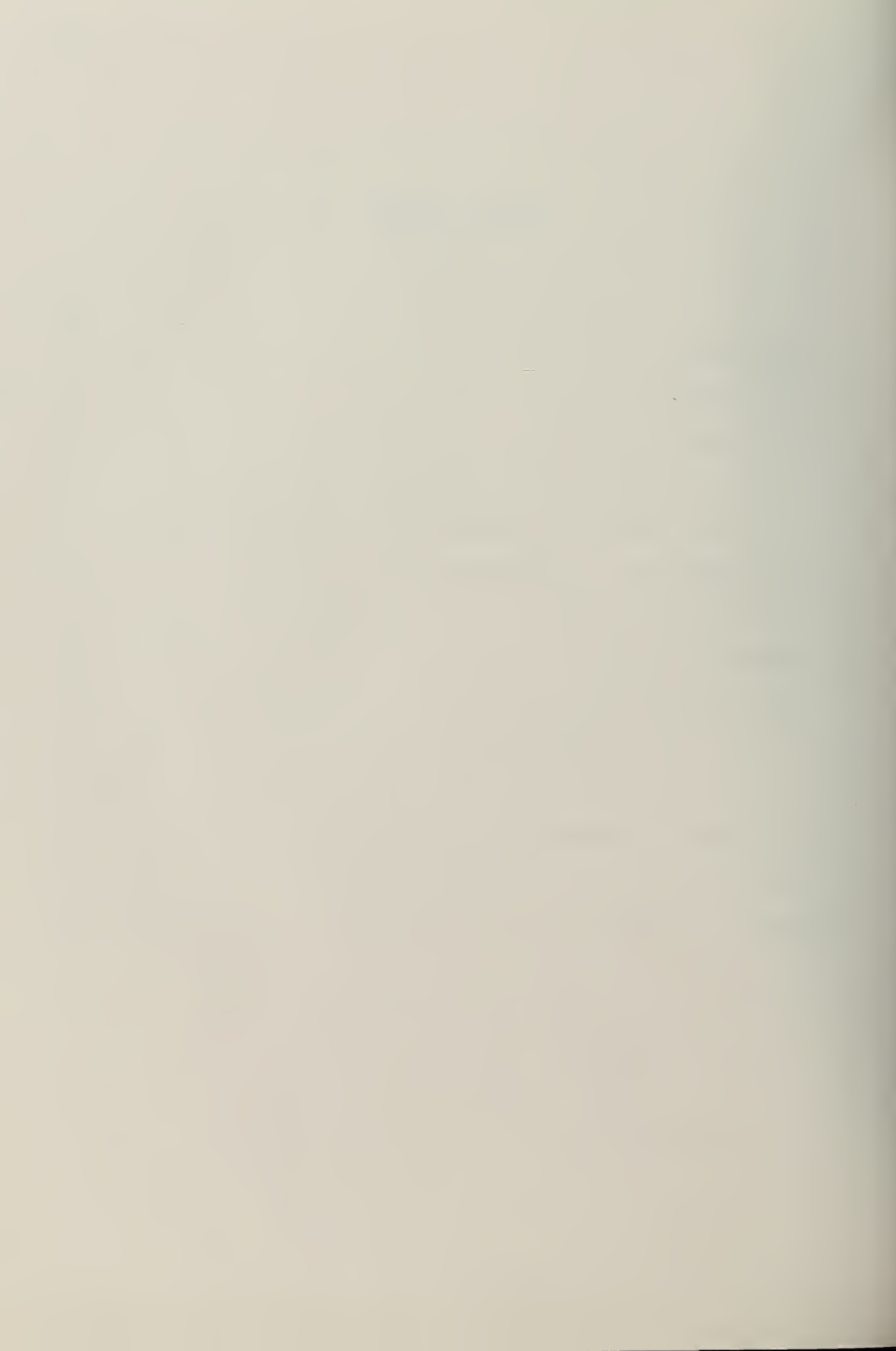


TABLE OF CONTENTS

	Page
1. Introduction.....	1
2. Riccati Equation.....	5
2.1 Power of the Method.....	7
2.1.1 Constant Coefficients.....	7
2.1.2 Variable Coefficients.....	8
2.2 Implementation Considerations.....	10
2.3 Initial Condition.....	12
3. Selection Procedures.....	13
3.1 Constant Coefficients.....	13
3.1.1 The Case with $\Delta < 0$	14
3.1.2 The Case with $\Delta > 0$	23
3.2 Variable Coefficients.....	24
4. Conclusion.....	27
References.....	28



1. Introduction

Recently, there has been a considerable interest in the representations of numbers other than the conventional positional notation for digital hardware calculations [1]; the concern here will be with the continued fractions. To facilitate the hardware implementation, we require that the coefficients of the continued fractions be integral powers of two. One important requirement for such a representation to be useful is that it should be possible to define a broad class of algorithms that are easily soluble. It was shown that a limited class of quadratics can be solved using this approach [1,2]. This was later extended to polynomials of degree larger than two [3]. An algorithm for logarithm was presented in [4]. The class of Riccati differential equations is closed under a bilinear transformation [5]. In this paper we show that a very large number of functions may be evaluated using the Riccati equation approach.

As a result of the restriction on the coefficients of the continued fractions, the selection of the coefficients, during the iterative evaluation of a function, becomes a difficult problem. We require that such a selection procedure be computationally "simple." It was shown that a simple selection procedure can be obtained for the algorithm for the quadratic equation [1,2]. This was later extended to the polynomials of degree larger than two [3]. Recently, we have shown that for an algorithm for logarithm, a simple selection procedure does not exist [4]. In this paper, we obtain similar negative results for many functions that can be evaluated using the Riccati differential equation.

An infinite continued fraction is represented by,

$$\frac{p_1}{q_1} + \frac{p_2}{q_2} + \dots$$

where p_i are known as the partial numerators and q_i are known as the partial denominators. The classical theory of continued fractions uses $p_i = 1$ and $q_i \in \mathbb{N}$ where \mathbb{N} is the set of natural numbers. We differ from this in that we require $p_i \in S_p$ and $q_i \in S_q$ such that S_p and S_q are finite and positive sets. If we let $p_{\min} = \min S_p$, $p_{\max} = \max S_p$, $q_{\min} = \min S_q$ and $q_{\max} = \max S_q$ then the smallest number, m , representable as an infinite continued fraction is the positive solution of the quadratic

$$m = \frac{p_{\min}}{q_{\max} + \frac{p_{\max}}{q_{\min} + m}}.$$

Similarly, the largest representable number, M , is the positive solution of the quadratic

$$M = \frac{p_{\max}}{q_{\min} + \frac{p_{\min}}{q_{\max} + M}}.$$

Let $m_{pq} = \frac{p}{q+M}$, $M_{pq} = \frac{p}{q+m}$ and $I_{pq} = [m_{pq}, M_{pq}]$ where $p \in S_p$ and $q \in S_q$. Note that, I_{pq} is a closed interval of the real numbers. It can be shown that [4] the set of numbers representable as infinite continued fractions, using finite and positive digit sets S_p and S_q , is complete iff

$$I_{S_p S_q} \triangleq \bigcup_{\substack{p \in S_p \\ q \in S_q}} I_{pq} = [m, M].$$

It can also be shown that if $S_p = \{1\}$ and $S_q \subseteq N$ then we have completeness only if $S_q = N$. But this conflicts with the requirement of finiteness.

Therefore, we will depart from the classical approach either by allowing fractions in S_q or by using a larger set of partial numerators or both.

Let the finite continued fraction $\frac{p_1}{q_1} + \frac{p_2}{q_2} + \dots + \frac{p_n}{q_n}$ be denoted by $\frac{P_n}{Q_n}$. Letting $P_0 = 0$, $Q_0 = 1$, $P_{-1} = 1$ and $Q_{-1} = 0$, we can evaluate such a fraction using the following recursions [6]:

$$P_{i+1} = p_{i+1} P_{i-1} + q_{i+1} P_i \quad i = 0, 1, \dots, n-1.$$

$$Q_{i+1} = p_{i+1} Q_{i-1} + q_{i+1} Q_i$$

Each iterative step of such an evaluation requires four multiplications and two additions. If we require that p_i 's and q_i 's are powers of two then these four multiplications can be reduced to simple shifts in binary arithmetic. We will, therefore, require that such be the case.

In the classical approach to function evaluation, a finite continued fraction with a few terms is used. Furthermore, the partial numerators and partial denominators are generally positive integral powers of the argument, x [7]. This will clearly require multiplications in an iterative step. Our approach requires that the partial numerators and denominators be simple powers of two. This implies that the complexity of function evaluation is transferred to a selection procedure which yields the value of the pair (p_i, q_i) at the i^{th} iterative step. Since such a selection procedure, in general, will be very complex (of the order of complexity of the function to be evaluated) and since it will be used in each iterative step, we are forced to use some approximation so as to

render it "simple." A "simple" selection procedure may use shift, add, subtract and comparison operations only. This leads us to a discussion of redundancy [2,4].

Given S_p and S_q , if we have completeness then the set of numbers representable as infinite continued fractions will be called a number system (NS). A number system is defined to be nonredundant if for all $p_1, p_2 \in S_p$ and $q_1, q_2 \in S_q$, $I_{p_1 q_1} \cap I_{p_2 q_2}$ is either null or is a singleton. A number system is redundant if it is not nonredundant. It can be easily shown that for a nonredundant number system, all but a countable set of numbers can be represented uniquely. Therefore, the use of any approximation in the selection procedure implies that we use a redundant number system.

Two approaches to function evaluation using continued fractions have been attempted. In the first approach, the function to be evaluated is $f(\underline{a}_0)$ where \underline{a}_0 is a vector of arguments and we expand $f(\underline{a}_i)$ using the following bilinear transformation:

$$f(\underline{a}_i) = \frac{p_{i+1}}{q_{i+1} + f(\underline{a}_{i+1})} .$$

We require that the vector of coefficients \underline{a}_{i+1} can be obtained from \underline{a}_i , p_{i+1} and q_{i+1} by means of "simple" recursions. A recursion is "simple" if it uses shift, addition and subtraction operations only. The algorithm for the solution of a quadratic equation [1] and the algorithm for logarithm [4] are members of this class.

In the second approach, we look for equations (algebraic or differential) which are closed under a bilinear transformation. All the

functions which are solutions to such equations can then be evaluated.

The Riccati differential equation is a member of this class.

In Section 2, we show that a very large number of functions can be evaluated using the Riccati equation approach. In Section 3, we show that no simple selection procedure exists for the functions discussed in Section 2.

2. Riccati Equation

Riccati equation can be written as:

$$y' + a(x)y^2 + b(x)y + c(x) = 0. \quad (2.1)$$

Let L be the set of all Riccati equations of this form. Wynn has shown that the set L is closed under the bilinear transformation $y = p/(q+z)$ where p, q are constants [5]. Starting with $\ell_0 \in L$, by a repeated application of the bilinear transformation, we can obtain a continued fraction expansion for the solution to the initial Riccati equation ℓ_0 .

Let ℓ_0 be given by: $y'_0 = a_0 y_0^2 + b_0 y_0 + c_0$, and let $y_0 = p_1/(q_1 + y_1)$. Let this transformation be called $T_1: L \rightarrow L$. $T_1(\ell_0) = \ell_1$ is given by $y'_1 + a_1 y_1^2 + b_1 y_1 + c_1 = 0$.

The recursion relations for the coefficients a_1, b_1, c_1 in terms of a_0, b_0, c_0 are,

$$\left. \begin{aligned} a_1 &= c_0/p_1, \\ b_1 &= b_0 + 2 c_0 q_1/p_1, \\ c_1 &= a_0 p_1 + b_0 q_1 + c_0 q_1^2/p_1 \end{aligned} \right\} \quad (2.2)$$

Note here that, we have changed the form of ℓ_0 to avoid negative signs in recursions (2.2). In general, let $\ell_{2m} = (y'_{2m} = a_{2m} y_{2m}^2 + b_{2m} y_{2m} + c_{2m})$ and let $\ell_{2m+1} = (y'_{2m+1} + a_{2m+1} y_{2m+1}^2 + b_{2m+1} y_{2m+1} + c_{2m+1} = 0)$. Assume that, $\ell_n = T_n T_{n-1} \dots T_1(\ell_0)$ has been obtained. Then the coefficients of $\ell_{n+1} = T_{n+1}(\ell_n)$ are given by the following recursions:

$$\left. \begin{aligned} a_{n+1} &= c_n / p_{n+1} \\ b_{n+1} &= b_n + 2 c_n q_{n+1} / p_{n+1} \\ c_{n+1} &= a_n p_{n+1} + b_n q_{n+1} + c_n q_{n+1}^2 / p_{n+1} \end{aligned} \right\} \quad (2.3)$$

As a result of these transformations, we have expanded y_0 to $n+1$ terms as follows:

$$y_0 = \frac{p_1}{q_1} + \frac{p_2}{q_2} + \dots + \frac{p_{n+1}}{q_{n+1} + y_{n+1}} \quad (2.4)$$

Let P_{n+1}/Q_{n+1} denote the finite continued fraction obtained by setting $y_{n+1} = 0$ in equation (2.4). If we assume that $|y_n| < M$ where M is a fixed constant then clearly, the fraction P_n/Q_n converges to y_0 . By setting, $P_0 = 0$, $Q_0 = 1$, $P_1 = p_1$ and $Q_1 = q_1$, the recursions for P_{n+1} and Q_{n+1} are [4]:

$$\left. \begin{aligned} P_{n+1} &= q_{n+1} P_n + p_{n+1} P_{n-1} \\ Q_{n+1} &= q_{n+1} Q_n + p_{n+1} Q_{n-1} \end{aligned} \right\} \quad (2.5)$$

Thus if we have a method to correctly choose p_n , q_n for every n then, we have an algorithm to solve the Riccati equation.

2.1 Power of the Method

We will now discuss the number of functions that can be obtained by the method of Riccati equation.

2.1.1 Constant Coefficients

Let us consider a subset L_0 of L such that

$L_0 = \{y' + ay^2 + by + c = 0 \mid a, b, c \in \mathbb{R}\}$ i.e., the set of all Riccati equations with constant coefficients. Consider $\ell_0 \in L_0$ given by,

$y'_0 = a_0 y_0^2 + b_0 y_0 + c_0$. Depending on the sign of $\Delta = b_0^2 - 4 a_0 c_0$, the solution $y_0(x)$ of ℓ_0 can be written as,

$$y_0(x) = \frac{\sqrt{-\Delta}}{2a_0} \left(\tan\left(\frac{\sqrt{-\Delta}}{2} x + A_0\right) - \frac{b_0}{\sqrt{-\Delta}} \right)$$

if $\Delta < 0$ and $a_0 \neq 0$;

$$y_0(x) = -\frac{1}{a_0 x} - \frac{b_0}{2a_0} + A_0 \quad \text{if } \Delta = 0, a_0 \neq 0;$$

$$y_0(x) = \frac{\sqrt{\Delta}}{-2a_0} \left(\tanh\left(\frac{\sqrt{\Delta}}{2} x + A_0\right) - \frac{b_0}{\sqrt{\Delta}} \right)$$

if $\Delta > 0, a_0 \neq 0$;

$$y_0(x) = A_0 e^{b_0 x} + c_0 x \quad \text{if } a_0 = 0.$$

Depending on the values of the coefficients a_0, b_0, c_0 and the initial condition $t_0 = y_0(0)$, many different functions may be evaluated as shown in the following table.

a_0	b_0	c_0	Δ	t_0	$y_0(x)$
1	0	1	-4	0	$\tan x$
-1	0	-1	-4	∞	$\cot x$
-1	0	0	0	∞	$1/x$
-1	0	1	4	∞	$\cot h x$
-1	0	1	4	0	$\tan h x$
0	± 1	0	>0	1	$e^{\pm x}$

Table 2.1

2.1.2 Variable Coefficients

Consider a subset L_1 of L so that,

$$L_1 = \{y' = a(x) y^2 + b(x) y + c(x) \mid a(x) = k(x) \bar{a},$$

$$b(x) = k(x) \bar{b}, c(x) = k(x) \bar{c}, \text{ and } \bar{a}, \bar{b}, \bar{c}$$

are constants}.

Recursions for \bar{a}_{n+1} , \bar{b}_{n+1} and \bar{c}_{n+1} can be derived from the recursions (2.3) and are as follows:

$$\left. \begin{aligned} \bar{a}_{i+1} &= \bar{c}_i / p_{i+1} \\ \bar{b}_{i+1} &= \bar{b}_i + 2\bar{c}_i q_{i+1} / p_{i+1} \\ \bar{c}_{i+1} &= \bar{a}_i p_{i+1} + \bar{b}_i q_{i+1} + \bar{c}_i q_{i+1}^2 / p_{i+1} \end{aligned} \right\} \quad (2.6)$$

Depending on the sign of $\bar{\Delta}_0 = \bar{b}_0^2 - 4a_0 \bar{c}_0$, the solution to ℓ_0 is given by:

$$y_0(x) = \frac{\sqrt{-\bar{\Delta}_0}}{2\bar{a}_0} \left(\tan\left(\frac{\sqrt{-\bar{\Delta}_0}}{2} \int k(x) dx + A_0\right) - \frac{\bar{b}_0}{\sqrt{-\bar{\Delta}_0}} \right)$$

$$\text{if } \bar{\Delta}_0 < 0, \bar{a}_0 \neq 0;$$

$$y_0(x) = -\frac{1}{\bar{a}_0 \int k(x) dx} - \frac{\bar{b}_0}{2\bar{a}_0} + A_0$$

$$\text{if } \bar{\Delta}_0 = 0, \bar{a}_0 \neq 0;$$

$$y_0(x) = -\frac{\sqrt{\bar{\Delta}_0}}{2\bar{a}_0} \left(\tanh\left(\frac{\sqrt{\bar{\Delta}_0}}{2} \int k(x) dx + A_0\right) - \frac{\bar{b}_0}{\sqrt{\bar{\Delta}_0}} \right)$$

$$\text{if } \bar{\Delta}_0 > 0, \bar{a}_0 \neq 0;$$

$$y_0(x) = A_0 e^{\frac{\bar{b}_0}{\bar{a}_0} \int k(x) dx} - \frac{\bar{c}_0}{\bar{b}_0} \quad \text{if } \bar{a}_0 = 0, \bar{b}_0 \neq 0;$$

and

$$y_0(x) = \bar{c}_0 \int k(x) dx + A_0 \quad \text{if } \bar{a}_0 = \bar{b}_0 = 0.$$

Clearly, a large class of functions can be evaluated with this method.

2.1.2.1 The Case With $\bar{\Delta}_0 = 0$

In this section, we will concentrate on a subset L_{10} of L_1 such that, $L_{10} = \{\ell \in L_1 \mid \bar{\Delta}_0 = 0\}$. Any $\ell \in L_{10}$ can be rewritten as: $y' = k(x)(a^*y + b^*)^2$ where, $a^* = \sqrt{\bar{a}}$, $b^* = a^*\left(\frac{\bar{b}}{2\bar{a}}\right)$. With this modification, we have reduced the number of coefficients from three to two. The recursions on a_n^* , b_n^* can now be written as follows:

$$\left. \begin{aligned} a_{n+1}^* &= b_n^* \sqrt{p_{n+1}}, \\ b_{n+1}^* &= (a_n^* p_{n+1} + b_n^* q_{n+1}) / \sqrt{p_{n+1}} \end{aligned} \right\} \quad (2.7)$$

The solution $\ell_0 \in L_{10}$ is given by,

$$\begin{aligned} y_0(x) &= \frac{1}{(a_0^*)^2 (A_0 - \int k(x) dx)} - \frac{b_0^*}{a_0^*} \quad \text{if } a_0^* \neq 0, \\ y_0(x) &= (b_0^*)^2 \int k(x) dx + A_0 \quad \text{if } a_0^* = 0. \end{aligned}$$

Note that, we can integrate the given function $k(x)$ by this method by setting $a_0^* = 0$ and $b_0^* = 1$.

2.2 Implementation Considerations

Let us assume that simple selection procedures are available for all the functions to be evaluated as detailed in section 2.1. We now give steps of an algorithm T which will evaluate these functions.

Algorithm T:

Step 1: [Initialize]

Set $P_0 \leftarrow 0$, $Q_0 \leftarrow 1$, $P_{-1} \leftarrow 1$, $Q_{-1} \leftarrow 0$;

Set initial values of coefficients according to the function to be evaluated;

Set $i \leftarrow 0$;

Step 2: [Select]

$(p_{i+1}, q_{i+1}) \leftarrow \text{Select}(x, \text{coefficients, function});$

Step 3: [Recursions]

$$P_{i+1} \leftarrow q_{i+1} P_i + p_{i+1} P_{i-1};$$

$$Q_{i+1} \leftarrow q_{i+1} Q_i + p_{i+1} Q_{i-1};$$

Recurse using equations (2.3), (2.6) or
(2.7) whichever is applicable.

Step 4: [Test]

After 'sufficient' number of iterations
GO TO Step 5; otherwise set $i \leftarrow i+1$,
and GO TO Step 2;

Step 5: [evaluate]

$$y_0(x) = f(x) = P_{i+1}/Q_{i+1};$$

END T;

In any such iterative algorithm, the number of iterations required and the execution time required per iteration are two important considerations. In each iteration, steps 2, 3 and 4 are executed. Clearly, step 2 and 3 require more attention. We can assume that if the procedure Select is known, it can be implemented in a combinational network and therefore, will require very little time. In step 3, we see that all the assignments are independent of each other and therefore, can be executed in parallel. Thus, given sufficient hardware, step 3 can be speeded up considerably. Each individual recursion requires additions (subtraction), multiplications and sometimes division also. Since multiplication and division are relatively slower operations, we would like to avoid them if possible. If we restrict the coefficients p_i and q_i to be integral powers of two these multiplications and divisions will be reduced to shifts, which is relatively a faster operation. If we use the recursions (2.7), then we further require that

$p_i = 1$ for $\forall i$. We will assume that $p_i \in S_p$, where $S_p = \{2^j | j \text{ is an integer}\}$ and $q_i \in S_q$ where, $S_q = \{2^j | j \text{ is an integer}\}$. Since the number of shifts available is finite we further require that, $S_p = \{2^j | \underline{J}_p \leq j \leq \bar{J}_p\}$ and $S_q = \{2^j | \underline{J}_q \leq j \leq \bar{J}_q\}$ where \bar{J}_p , \underline{J}_p , \bar{J}_q and \underline{J}_q are fixed integers.

The number of iterations to be carried out can be decided on the basis of allowable error in the result.

2.3 Initial Condition

Associated with the solution of any differential equation, there are one or more arbitrary constants which are evaluated using the boundary conditions imposed. Depending on the Function $f(x)$ to be evaluated, we choose a particular $\ell_0 \in L$ (and the corresponding coefficient values) and the associated initial condition $y_0(0) = t_0$ so that $y_0(x) = f(x)$. Clearly, the initial condition on ℓ_i ($i \geq 1$) is dependent on t_0 . In particular,

$$y_{n-1} = p_n / (q_n + y_n) \text{ which implies,}$$

$$t_{n-1} = p_n / (q_n + t_n) \text{ which implies,} \quad (2.9)$$

$$t_n = p_n / t_{n-1} - q_n$$

As we will see in Chapter 3, t_n is needed as an argument in a selection procedure for p_{n+1} and q_{n+1} . Therefore, we need to evaluate t_n in every iteration. This, however, implies that a division be carried out. We can avoid the division by the following technique.

Let $t_n = d_n / e_n$ then, from equation (2.8),

$$\frac{d_n}{e_n} = -q_n + \frac{p_n e_{n-1}}{d_{n-1}}$$

From which,

$$\left. \begin{aligned} d_n &= p_n e_{n-1} - q_n d_{n-1} \\ e_n &= d_{n-1} \end{aligned} \right\} \quad (2.9)$$

and $d_0 = t_0$ and $e_0 = 1$.

If the selection procedure can choose with the help of d_n and e_n (does not explicitly require t_n) then we have solved our problem. Now in step 3 of algorithm T, we have to carry out recursions (2.9) as well.

3. Selection Procedures

We have seen that the form of the solution to a Riccati equation depends on the sign of the discriminant Δ . It is also clear that the selection procedure will be different for different forms of the solution, i.e., depending on the sign of Δ . Therefore, if Δ remains invariant under the bilinear transformation then hopefully the same selection procedure can be used consistently during the iterative evaluation of a function. It can be easily seen that this is indeed the case, i.e., $\Delta_i = \Delta_{i-1} = \dots = \Delta_0$.

In Section 3.1, we consider selection procedures for Riccati equations with constant coefficients, and in Section 3.2, we consider the more general case of variable coefficients.

3.1 Constant Coefficients

We will consider two subcases separately depending upon the value of the discriminant Δ .

3.1.1 The Case With $\Delta < 0$

Consider ℓ such that $y_i' = j(a_i y_i^2 + b_i y_i + c_i)$ where $a_i \neq 0$ and $j = 1$ if i is even and -1 otherwise. The solution to this equation is given by,

$$y_i(x) = \frac{j\sqrt{-\Delta}}{2a_i} \left[\tan \left(\frac{\sqrt{-\Delta}}{2} x + A_i \right) - \frac{jb_i}{\sqrt{-\Delta}} \right] \dots \quad (3.1)$$

If we let the initial condition be, $y_i(0) = d_i/e_i$ then we can evaluate the arbitrary constant A_i by substituting the initial condition in equation (3.1). Thus,

$$\frac{d_i}{e_i} = j \frac{\sqrt{-\Delta}}{2a_i} \left(\tan(A_i) - \frac{jb_i}{\sqrt{-\Delta}} \right) \text{ from which,}$$

$$A_i = j \arctan \left(\frac{2a_i d_i + b_i e_i}{e_i \sqrt{-\Delta}} \right).$$

Substituting in (3.1), we get,

$$y_i(x) = j \frac{\sqrt{-\Delta}}{2a_i} \left[\frac{\tan \left(\frac{\sqrt{-\Delta}}{2} x \right) + j \frac{2a_i d_i + b_i e_i}{e_i \sqrt{-\Delta}}}{1 - j \tan \left(\frac{\sqrt{-\Delta}}{2} x \right) \frac{2a_i d_i + b_i e_i}{e_i \sqrt{-\Delta}}} \right] - \frac{b_i}{2a_i}$$

$$= \frac{j\sqrt{-\Delta} \tan \left(\frac{\sqrt{-\Delta}}{2} x \right) \cdot e_i \sqrt{-\Delta} + \sqrt{-\Delta} (2a_i d_i + b_i e_i) - b_i (e_i \sqrt{-\Delta} - j \dots}{2a_i (e_i \sqrt{-\Delta} - j \tan \left(\frac{\sqrt{-\Delta}}{2} x \right) (2a_i d_i + b_i e_i))}$$

$$\begin{aligned}
&= \frac{j \tan \left(\frac{\sqrt{-\Delta}}{2} x \right) [-e_i \Delta + b_i (2a_i d_i + b_i e_i)] + \sqrt{-\Delta} (2a_i d_i)}{2a_i e_i \sqrt{-\Delta} - j \tan \left(\frac{\sqrt{-\Delta}}{2} x \right) (2a_i d_i + b_i e_i)} \\
&= \frac{j r_i u + (\sqrt{-\Delta}) d_i}{(\sqrt{-\Delta}) e_i - j h_i u} \tag{3.2}
\end{aligned}$$

where $r_i = 2c_i e_i + b_i d_i$, $h_i = 2a_i d_i + b_i e_i$ and $u = \tan \left(\frac{\sqrt{-\Delta}}{2} x \right)$. It is clear that the process of selection will involve r_i , h_i , d_i and e_i but not a_i , b_i , and c_i . Therefore, if we could obtain recursions for r_i and h_i which are free of a_i , b_i and c_i then we will avoid the computation of a_i , b_i and c_i . We will now derive the recursions for h_i and r_i using the recursions for a_i , b_i and c_i and a slightly more general form of recursions for d_i and e_i than those used in (2.9). The recursions for d_i and e_i are as follows:

$$d_{i+1} = k_{i+1} (p_{i+1} e_i - q_{i+1} d_i)$$

and

$$e_{i+1} = k_{i+1} d_i.$$

Now,

$$\begin{aligned}
h_{i+1} &= 2a_{i+1} d_{i+1} + b_{i+1} e_{i+1} \\
&= 2(c_i/p_{i+1}) k_{i+1} (p_{i+1} e_i - q_{i+1} d_i) + \\
&\quad (b_i + 2c_i q_{i+1}/p_{i+1}) k_{i+1} d_i \\
&= 2k_{i+1} c_i e_i + k_{i+1} b_i d_i \\
&= k_{i+1} r_i.
\end{aligned}$$

In a similar way, we can obtain the recursion for r_i . As a result, the set of recursions that we will use is as follows:

$$\left. \begin{aligned} h_{i+1} &= k_{i+1} r_i \\ r_{i+1} &= k_{i+1} (p_{i+1} h_i + q_{i+1} r_i) \\ d_{i+1} &= k_{i+1} (p_{i+1} e_i - q_{i+1} d_i) \\ e_{i+1} &= k_{i+1} d_i \end{aligned} \right\} \quad (3.3)$$

The condition for the selection of a (p, q) pair is given by:

$y_i(x) \in I_{pq}$. In other words, the selection condition is: If

$$m_{pq} \leq \frac{j r_i u + \sqrt{-\Delta}}{\sqrt{-\Delta} e_i - j h_i u} \leq M_{pq} \text{ then choose } (p, q). \text{ Note that, we cannot}$$

use this condition directly since u is an unknown, therefore, we would like to rewrite the selection condition as follows:

$$\arctan(\operatorname{ARG}_i(m_{pq})) \leq \frac{\sqrt{-\Delta} j x}{2} \leq \arctan(\operatorname{ARG}_i(M_{pq})) \quad (3.4)$$

where

$$\operatorname{ARG}_i(s) = \frac{\sqrt{-\Delta} e_i s - \sqrt{-\Delta} d_i}{r_i + s h_i}$$

Note that such a rewriting is valid if both of the following conditions are satisfied: (1) $\operatorname{ARG}_i(s)$ is a monotone-increasing function of s , and (2) $\arctan(z)$ is a monotone-increasing function of z . Since condition (2) is already known to be satisfied, we only have to verify the condition (1). To do this, note that,

$$\begin{aligned}\frac{\partial \text{ARG}_i(s)}{\partial s} &= \frac{(r_i + h_i s)(\sqrt{-\Delta} e_i) - h_i \sqrt{-\Delta}(e_i s - d_i)}{(r_i + h_i s)^2} \\ &= \sqrt{-\Delta}(r_i e_i + h_i d_i) / (r_i + h_i s)^2.\end{aligned}$$

Now

$$\begin{aligned}r_{i+1} e_{i+1} + h_{i+1} d_{i+1} &= k_{i+1} (p_{i+1} h_i + q_{i+1} r_i) k_i d_i + \\ &\quad k_{i+1}^2 r_i (p_{i+1} e_i - q_{i+1} d_i) \\ &= k_{i+1}^2 (p_{i+1} h_i d_i + p_{i+1} r_i e_i) \\ &= p_{i+1} k_{i+1}^2 (r_i e_i + h_i d_i).\end{aligned}$$

Therefore,

$$r_i e_i + h_i d_i = \left(\prod_{j=1}^i (p_j k_j^2) \right) (r_0 e_0 + h_0 d_0).$$

Therefore, $\text{ARG}_i(s)$ is a monotone-increasing function of s provided $r_0 e_0 + h_0 d_0 > 0$. Observe that there is no loss of generality in assuming that $r_0 e_0 + h_0 d_0 > 0$. Since if $r_0 e_0 + h_0 d_0 < 0$ then $\text{ARG}_i(s)$ will be a monotone-decreasing function of s and we can turn the inequality (3.4) around and follow very similar arguments. Also note that the condition $r_0 e_0 + h_0 d_0 = 0$ will not occur, since this implies that either t_0 (the initial condition) is complex or $d_0 = e_0 = 0$ or $a_0 = 0$.

In theory, the selection condition (3.4) can be used to select the (p, q) pair during each iterative step, but the amount of computation involved is clearly excessive. We note that in order to compute a boundary of a selection region, $\arctan(\text{ARG}_i(s))$ needs to be computed and there are

as many as $|S_p| \times |S_q|$ selection regions. It is, therefore, clear that we would like to use an approximation to $\arctan(\text{ARG}_i(s))$ which is "easy" enough to compute from the available coefficients h_i, r_i, d_i, e_i and the known value of s . We note that the use of an approximation in the selection procedure implies the use of redundancy in the digit sets since otherwise we cannot guarantee correct selection.

With the use of redundancy, there will be regions in which more than one (p, q) pair can be chosen. Define $I_{p_1 q_1} < I_{p_2 q_2}$ if there exists $f \in I_{p_1 q_1}$ such that for all $g \in I_{p_2 q_2}, f < g$. A pair (p_2, q_2) is said to be right-adjacent to a pair (p_1, q_1) if $I_{p_1 q_1} < I_{p_2 q_2}$ and for all (p_3, q_3) such that $I_{p_1 q_1} < I_{p_3 q_3}, I_{p_2 q_2} \leq I_{p_3 q_3}$. A similar definition of left-adjacency can be given. Given a pair (p_1, q_1) its left-adjacent pair (p_2, q_2) and the right-adjacent pair (p_3, q_3) , the following holds: If $f \in I_{p_1 q_1} \cap I_{p_3 q_3}$ then we can choose (p_3, q_3) or (p_1, q_1) , if $f \in I_{p_1 q_1} \cap I_{p_2 q_2}$ then we can choose (p_1, q_1) or (p_2, q_2) and if $f \in I_{p_1 q_1} - (I_{p_1 q_1} \cap I_{p_3 q_3}) - (I_{p_1 q_1} \cap I_{p_2 q_2})$ then we must choose the pair (p_1, q_1) . We note that the existence of selection overlap regions such as $I_{p_1 q_1} \cap I_{p_3 q_3}$ allows us to use an approximation in the selection procedure. Let us denote the approximate value of $\arctan(\text{ARG}_i(s))$ by $\text{AT}_i(s)$, then the selection rule to be used can be specified by:

$$\text{If } \text{AT}_i(z_1) \leq \frac{\sqrt{-\Delta}}{2} \leq \text{AT}_i(z_2) \text{ then choose } (p_1, q_1) \quad (3.5)$$

where $z_1 \in I_{p_1 q_1} \cap I_{p_3 q_3}$ and $z_2 \in I_{p_1 q_1} \cap I_{p_2 q_2}$. Note that z_1, z_2 will

now be a boundary between adjacent selection regions and therefore the selection of the (p, q) pair will now be unique. In order to guarantee correct selection using condition (3.5), we have to show that the region specified by condition (3.5) is a subset of the region specified by the condition (3.4). From this, we can say that the maximum error allowable in the computation of $\arctan(\text{ARG}_i(s))$, denoted by E_i , is given by:

$$E_i = \text{Max} [\arctan(\text{ARG}_i(M_{p_1 q_1})) - \text{AT}_i(z_2), \\ \text{AT}_i(z_1) - \arctan(\text{ARG}_i(m_{p_1 q_1}))].$$

In other words, we can find s_1 and s_2 ($s_2 > s_1$) such that,

$$E_i \leq \arctan(\text{ARG}_i(s_2)) - \arctan(\text{ARG}_i(s_1)).$$

Now we note that, $\arctan(z)$ satisfies the Lipschitz condition, i.e.,

$$|\arctan(z_2) - \arctan(z_1)| \leq L|z_2 - z_1|$$

for $L > 0$ and $L \leq \Pi$. Therefore,

$$E_i \leq L (\text{ARG}_i(s_2) - \text{ARG}_i(s_1)). \quad (3.6)$$

Now,

$$H_i = \text{ARG}_i(s_2) - \text{ARG}_i(s_1) \\ = \frac{\sqrt{-\Delta} (e_i s_2 - d_i)}{(r_i + h_i s_2)} - \frac{\sqrt{-\Delta} (e_i s_1 - d_i)}{(r_i + h_i s_1)} \\ = \frac{(r_i e_i + h_i d_i) (\sqrt{-\Delta}) (s_2 - s_1)}{(s_1 h_i + r_i) (s_2 h_i + r_i)}.$$

Using an expression derived for $r_i e_i + h_i d_i$ earlier, we have,

$$H_i = \frac{\sqrt{-\Delta} (s_2 - s_1) \left(\prod_{j=1}^i p_j k_j^2 \right) (r_0 e_0 + h_0 d_0)}{(s_1 h_i + r_i) (s_2 h_i + r_i)} \quad (3.7)$$

We are now interested in eliminating h_i and r_i from the expression of H_i .

Towards this end, we will show that,

$$r_i = r_0 K_i Q_i + h_0 K_i P_i$$

where

$$K_i = \prod_{j=1}^i (k_j).$$

We proceed to prove this result by induction on i . Since $P_0 = 0$, $Q_0 = 1$

and $K_0 = 1$, we have $r_0 = r_0 \cdot 1 \cdot 1 + h_0 \cdot 1 \cdot 0 = r_0$. Now recursions (3.3),

we have,

$$r_1 = k_1 (r_0 q_1 + h_0 p_1) = r_0 K_1 Q_1 + h_0 K_1 P_1.$$

Now assume that the required result is true for r_j . For $j \leq i$. Again from recursions (3.3),

$$\begin{aligned} r_{i+1} &= k_{i+1} (p_{i+1} h_i + q_{i+1} r_i) \\ &= k_{i+1} (p_{i+1} k_i r_{i-1} + q_{i+1} r_i) \\ &= k_{i+1} (p_{i+1} k_i (r_0 K_{i-1} Q_{i-1} + h_0 K_{i-1} P_{i-1}) + q_{i+1} (r_0 K_i Q_i + h_0 K_i P_i)) \\ &= r_0 K_{i+1} (p_{i+1} Q_{i-1} + q_{i+1} Q_i) + h_0 K_{i+1} (p_{i+1} P_{i-1} + q_{i+1} P_i) \\ &= r_0 K_{i+1} Q_{i+1} + h_0 K_{i+1} P_{i+1}. \end{aligned}$$

Thus, we have the required result. It follows from this that

$$h_i = k_i r_{i-1} = K_i (r_0 Q_{i-1} + h_0 P_{i-1})$$

Now substituting these expressions for h_i and r_i in the equation (3.7), we have,

$$H_i = \frac{\left(\prod_{j=1}^i p_j \right) K_i^2 (r_0 e_0 + h_0 d_0) \sqrt{-\Delta} (s_2 - s_1)}{K_i^2 [s_1 (r_0 Q_{i-1} + h_0 P_{i-1}) + r_0 Q_i + h_0 P_i] * [s_2 (r_0 Q_{i-1} + h_0 P_{i-1}) + r_0 Q_i + h_0 P_i]}.$$

Substituting this in the expression (3.6), we have,

$$E_i \leq \frac{\left(\prod_{j=1}^i p_j \right) L (r_0 e_0 + h_0 d_0) \sqrt{-\Delta} (s_2 - s_1)}{[s_1 (r_0 Q_{i-1} + h_0 P_{i-1}) + r_0 Q_i + h_0 P_i] [s_2 (r_0 Q_{i-1} + h_0 P_{i-1}) + r_0 Q_i + h_0 P_i]}.$$

Now we consider two cases, depending upon the value of r_0 . If $r_0 \neq 0$ then we have,

$$E_i \leq B_1 \frac{\left(\prod_{j=1}^i p_j \right)}{Q_i Q_{i-1}} \quad (3.8)$$

since $P_i, Q_i, P_{i-1}, Q_{i-1}, s_1, s_2$ are all > 0 and where

$$B_1 = L \left(\frac{r_0 e_0 + h_0 d_0}{r_0^2} \right) \left(\frac{s_2 - s_1}{s_2} \right) \sqrt{-\Delta}.$$

On the other hand if $r_0 = 0$

$$E_i \leq \frac{\left(\prod_{j=1}^i p_j \right) L h_0 d_0 \sqrt{-\Delta} (s_2 - s_1)}{h_0^2 (s_1 P_{i-1} + P_i) (s_2 P_{i-1} + P_i)}$$

$$\leq \frac{\prod_{j=1}^i p_j}{P_i P_{i-1}} \left(\frac{s_2 - s_1}{s_2} \right) \sqrt{-\Delta} \frac{d_0}{h_0} \quad (3.9)$$

We will now obtain a bound on $P_i P_{i-1}$ in terms of $Q_i Q_{i-1}$. A well known property of the convergents of an infinite continued fraction, f , can be written as [6]:

$$\frac{P_0}{Q_0} \leq \frac{P_2}{Q_2} \leq \dots \leq f \leq \dots \leq \frac{P_3}{Q_3} \leq \frac{P_1}{Q_1}.$$

Therefore, if i is odd, $\frac{P_i}{Q_i} \geq m$. If $i \geq 2$ is even, $\frac{P_i}{Q_i} \geq \frac{P_2}{Q_2} \geq \frac{p_{\min}}{q_{\max} + \frac{p_{\max}}{q_{\min}}}$.

Therefore,

$$\frac{P_i}{Q_i} \cdot \frac{P_{i-1}}{Q_{i-1}} \geq \frac{m p_{\min}}{q_{\max} + \frac{p_{\max}}{q_{\min}}}$$

Substituting this in (3.9), we have,

$$E_i \leq \frac{\prod_{j=1}^i p_j}{Q_i Q_{i-1}} \cdot B_2 \quad (3.10)$$

where $B_2 = \frac{s_2 - s_1}{s_2} \cdot \sqrt{-\Delta} \cdot \frac{d_0}{h_0} \cdot \frac{m p_{\min}}{q_{\max} + \frac{p_{\max}}{q_{\min}}}$. From (3.9) and (3.10), we have,

$$E_i \leq B \frac{\prod_{j=1}^i p_j}{Q_i Q_{i-1}}$$

where $B = B_1$ if $r_0 \neq 0$ and B_2 otherwise. Note that B is a fixed, finite and bounded constant independent of the value of i . The factor

$\frac{\prod_{j=1}^i p_j}{Q_i Q_{i-1}}$ can be interpreted as the error in the solution, since

it equals the difference in values of the successive convergents P_{i-1}/Q_{i-1} and P_i/Q_i [6]. Therefore, if we demand linear convergence then we must have,

$$\frac{\prod_{j=1}^i p_j}{Q_i Q_{i-1}} = \text{constant} \cdot \alpha^{-i}$$

for a small positive constant and some $\alpha > 1$. As a result, we have,

$$E_i \leq B' \cdot \alpha^{-i}.$$

But this implies that the computation of $\arctan(\text{ARG}_i(s))$ must be carried out to nearly the same precision as that of the desired precision of the function being evaluated. Thus we conclude that we cannot obtain a computationally simple selection procedure for the functions that can be evaluated using the Riccati equation with constant coefficients and $\Delta < 0$.

3.1.2 The Case With $\Delta > 0$

Consider the following Riccati equation:

$$y'_i = j(a_i y_i^2 + b_i y_i + c_i)$$

such that $\Delta = \Delta_i > 0$ and $j = 1$ if i is even and -1 otherwise. The solution to this equation can be written as,

$$y_i(x) = \frac{\sqrt{\Delta}}{2a_i} \coth\left(\frac{jx\sqrt{\Delta}}{2} + A_i\right) - \frac{b_i}{2a_i} \quad (3.11)$$

where A_i is an arbitrary constant of integration. Using the initial condition $y_i(0) = t_i = d_i/e_i$, we obtain $\tanh A_i = \frac{\sqrt{\Delta} e_i}{2a_i d_i + b_i e_i}$. For the sake

of brevity, we let $h_i = 2a_i d_i + b_i e_i$ and after substituting for A_i in (3.11), we get,

$$y_i(x) = \frac{1}{2a_i} \left\{ \frac{j \Delta e_i \tanh\left(\frac{\sqrt{\Delta}x}{2}\right) - j b_i h_i \tanh\left(\frac{\sqrt{\Delta}x}{2}\right) - \sqrt{\Delta} 2a_i d_i}{j h_i \tanh\left(\frac{\sqrt{\Delta}x}{2}\right) - \sqrt{\Delta} e_i} \right\}$$

From which, we get,

$$j \tanh\left(\frac{\sqrt{\Delta}x}{2}\right) = \frac{\sqrt{\Delta} (y_i e_i - d_i)}{(y_i h_i + r_i)} \quad (3.12)$$

where $r_i = b_i d_i + 2c_i e_i$. From equation (3.11), we note that if $e_0 = 1$, $d_0 = 0$, $h_0 = 0$ and $r_0 = \sqrt{\Delta}$ then $y_0(x) = \tanh\left(\frac{\sqrt{\Delta}x}{2}\right)$. If $e_0 = 0$, $d_0 = 1$, $h_0 = -\sqrt{\Delta}$ and $r_0 = 0$ then $y_0(x) = \coth\left(\frac{\sqrt{\Delta}x}{2}\right)$. If $c_0 = 0$ and $a_0 = 0$ then we have $y_0(x) = A_0 e^{b_0 x}$.

From the form of the equation (3.12) and the definitions of r_i and h_i , it is clear that we can follow the same arguments as in Section 3.1.1 and prove that a computationally simple selection procedure cannot be obtained in the case that $\Delta > 0$ or $a_0 = 0$. Thus we have shown the negative results for the Riccati equation with constant coefficients.

3.2 Variable Coefficients

We will only consider the case with $\bar{\Delta}_0 = 0$, i.e., we consider the subset L_{10} of L . Consider the equation

$$y_i' = j k(x) (a_i y + b_i)^2 \quad (3.13)$$

where $j = 1$ if i is even and zero otherwise. We will use the following set of recursions:

$$\left. \begin{aligned}
 a_{i+1} &= b_i \sqrt{p_{i+1}} \\
 b_{i+1} &= a_i \sqrt{p_{i+1}} + b_i q_{i+1} \sqrt{p_{i+1}} \\
 d_{i+1} &= e_i \sqrt{p_{i+1}} - d_i q_{i+1} \sqrt{p_{i+1}} \\
 e_{i+1} &= d_i \sqrt{p_{i+1}}
 \end{aligned} \right\} \quad (3.14)$$

The solution to this equation is given by:

$$y_i(x) = \frac{d_i + j(g(x)-g(0)) b_i (a_i b_i + d_i e_i)}{e_i - j(g(x)-g(0)) a_i (a_i b_i + d_i e_i)} \quad (3.15)$$

where $g(x) = \int k(x) dx$. To simplify the equation (3.15), we can easily prove by induction on i , that

$$a_i b_i + d_i e_i = a_0 b_0 + d_0 e_0 \triangleq r_0 .$$

Note that (using the recursions 3.14),

$$\begin{aligned}
 & a_{i+1} b_{i+1} + d_{i+1} e_{i+1} \\
 &= b_i \sqrt{p_{i+1}} (e_i \sqrt{p_{i+1}} - d_i q_{i+1} \sqrt{p_{i+1}}) + \\
 & \quad (a_i \sqrt{p_{i+1}} + b_i q_{i+1} \sqrt{p_{i+1}}) d_i \sqrt{p_{i+1}} \\
 &= a_i d_i + b_i e_i .
 \end{aligned}$$

Using this, we get

$$y_i(x) = \frac{d_i + j(g(x)-g(0)) b_i r_0}{e_i + j(g(x)-g(0)) a_i r_0} .$$

The selection condition can now be written as: If

$$m_{pq} \leq \frac{d_i + j(g(x)-g(0)) b_i r_0}{e_i + j(g(x)-g(0)) a_i r_0} \leq M_{pq} \text{ then choose } (p,q).$$

Since $g(x)$ is the unknown we want to transform the selection condition to:

$$\begin{aligned} g^{-1} \left\{ \frac{j(M_{pq} e_i - d_i + j M_{pq} g(0) a_i r_0)}{b_i r_0 + M_{pq} a_i r_0} \right\} &\leq x \\ &\leq g^{-1} \left\{ \frac{j(m_{pq} e_i - d_i + j m_{pq} g(0) a_i r_0)}{r_0 (b_i + m_{pq} a_i)} \right\} \end{aligned} \quad (3.15)$$

But this transformation is valid provided, $ARG_i(s)$ is a monotone-increasing function of s and $g^{-1}(z)$ is a monotone-increasing function of z . Note that,

$$ARG_i(s) = \frac{s e_i - d_i + j s g(0) a_i r_0}{r_0 (b_i + s a_i)}.$$

Therefore,

$$\begin{aligned} \frac{\partial ARG_i(s)}{\partial s} &= \frac{r_0 (b_i + s a_i) (e_i + j g(0) a_i r_0) - (s e_i - d_i + j s g(0) a_i r_0) r_0 a_i}{r_0^2 (b_i + s a_i)^2} \\ &= \frac{1 + j g(0) a_i b_i}{(b_i + s a_i)^2} \end{aligned}$$

For simplicity, we assume $g(0) = 0$ then clearly, $ARG_i(s)$ is a monotone-increasing function of s . We also assume that $g^{-1}(z)$ is a monotone-increasing function of z . If it is a monotone-decreasing then we can turn the inequality (3.15) around and similar arguments can be carried out.

The inequality (3.15) can be split up into two parts depending upon the value of i . We will only consider the case when i is even, the

other case being very similar. Then the selection condition is:

$$g^{-1}(\text{ARG}_i(m_{pq})) \leq x \leq g^{-1}(\text{ARG}_i(M_{pq})).$$

Now since $g^{-1}(\text{ARG}_i(s))$ is difficult to compute in general, therefore, we would like to use an approximation. The maximum error allowable in such an approximation can be written as,

$$E_i = g^{-1}(\text{ARG}_i(s_2)) - g^{-1}(\text{ARG}_i(s_1))$$

where $m \leq s_1 < s_2 \leq M$. Now we assume that g^{-1} satisfies the Lipschitz condition with "small" value of the Lipschitz constant L . Then

$$E_i \leq L[\text{ARG}_i(s_2) - \text{ARG}_i(s_1)] \quad (3.16)$$

Now,

$$\begin{aligned} H_i &= \text{ARG}_i(s_2) - \text{ARG}_i(s_1) \\ &= \frac{s_2 e_i - d_i}{r_0(b_i + s_2 a_i)} - \frac{s_1 e_i - d_i}{r_0(b_i + s_1 a_i)} \\ &= \frac{s_2 - s_1}{(b_i + s_2 a_i)(b_i + s_1 a_i)} \end{aligned}$$

From this point onwards, we can follow a procedure similar to Section 3.1 to obtain a similar negative result.

4. Conclusion

Recently, there has been some interest in the use of continued fractions for digital hardware calculations. We require that the coefficients of the continued fractions be integral powers of two. As a result, the selection of coefficients during the iterative evaluation of a function becomes a difficult problem. We have shown that practical selection procedures do not exist for most functions evaluated using the Riccati equation approach.

References

- [1] Robertson, J. E. and K. S. Trivedi, "The Status of Investigations into Computer Hardware Design Based on the Use of Continued Fractions," IEEE Transactions on Computers, Vol. C-22, No. 6, June 1973, pp. 555-560.
- [2] Trivedi, K. S., "An Algorithm for the Solution of a Quadratic Equation Using Continued Fractions," M.S. Thesis, University of Illinois, Urbana, June 1972; also Department of Computer Science Report #525.
- [3] Bracha, A., "A Method for Solving Polynomial Equations by Continued Fractions," IEEE Transactions on Computers, Vol. C-23, No. 10, October 1974, pp. 1093-1097.
- [4] Trivedi, K. S., "On a Negative Result Regarding the Use of Continued Fractions for Digital Computer Arithmetic," Department of Computer Science Report #693, University of Illinois, Urbana, January, 1975.
- [5] Wynn, P., "On Some Recent Developments in the Theory and Application of Continued Fractions," Journal SIAM on Num. Anal., Vol. 1, pp. 177-197, 1964.
- [6] Wall, H., "Analytic Theory of Continued Fractions," Van Nostrand, Princeton, New Jersey, 1950.
- [7] Khovanskii, A. N., "The Application of Continued Fractions," P. Nordhoff, N. V. - Groningen - The Netherlands, 1963.

BIBLIOGRAPHIC DATA SHEET	1. Report No. UIUCDCS-R-75-721	2.	3. Recipient's Accession No.
4. Title and Subtitle Further Negative Results Regarding the Use of Continued Fractions for Digital Computer Arithmetic		5. Report Date May 1975	
7. Author(s) Kishor Shridharbhai Trivedi		6.	
9. Performing Organization Name and Address Department of Computer Science University of Illinois Urbana, Illinois 61801		8. Performing Organization Rept. No.	
12. Sponsoring Organization Name and Address National Science Foundation Washington, D.C		10. Project/Task/Work Unit No.	
		11. Contract/Grant No. NSF DCR 73-07998	
		13. Type of Report & Period Covered	
15. Supplementary Notes		14.	
16. Abstracts Recently, there has been some interest in the use of continued fractions for digital hardware calculations. We require that the coefficients of the continued fractions be integral powers of two. As a result, the selection of coefficients during the iterative evaluation of a function becomes a difficult problem. In this paper, we show that no practical selection procedure exists for most functions evaluated using the Riccati equation approach.			
17. Key Words and Document Analysis. 17a. Descriptors Bilinear Transformation Completeness Computer Arithmetic Continued Fractions Hardware Redundancy Riccati Equation Selection Procedure			
17b. Identifiers/Open-Ended Terms			
17c. COSATI Field/Group			
18. Availability Statement		19. Security Class (This Report) UNCLASSIFIED	21. No. of Pages
		20. Security Class (This Page) UNCLASSIFIED	22. Price

JUN 26 1975



UNIVERSITY OF ILLINOIS-URBANA



3 0112 047424251